UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/725,728	12/02/2003	Kwasi Addo Asare	RSW920030191US1 (123)	3074

46320 7590 11/28/2007
CAREY, RODRIGUEZ, GREENBERG & PAUL, LLP
STEVEN M. GREENBERG
950 PENINSULA CORPORATE CIRCLE
SUITE 3020
BOCA RATON, FL 33487

EXAMINER	
KHATRI, ANIL	

ART UNIT	PAPER NUMBER
2191	

MAIL DATE	DELIVERY MODE
11/28/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/725,728	**Applicant(s)** ASARE ET AL.	
	Examiner Anil Khatri	**Art Unit** 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 September 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-17* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-17 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Hempstead et al* USPN 7,092,958 in view of *Goiffon et al* USPN 6,785,882.

Regarding claims 1 and 5

Hampstead et al teaches,

a repository of semantic models for interdependent ones of application components (column 10, lines 5-18, a third translation entails a conversion of MDBC Semantic Model to ODBC Catalog Information, identified by reference numeral 72 in FIG. 6. The ODBC standard specifies that catalog information (i.e., meta-data) should be returned in a specific manner via specific functions. The MDBC component returns its semantic model (i.e., meta-data) in a proprietary structure format defined in its Object Request Broker (ORB) Interface Definition Language (IDL) file. In order to convert the meta-data from the MDBC semantic model structure to the ODBC catalog structure, several algorithms were created. Multiple algorithms were needed because ODBC requires multiple method calls to return all the information that MDBC returns in its semantic model structure);

a mapping of individual listings in said semantic models to target platform specific installation instructions (column 6, lines 52-64, FIG. 5 illustrates the architecture used by the global database integration system, or Merlin 40. More particularly, Merlin 40 relies on object model data within layer 34 and data source mappings within layer 36 in order to define the specific domain that it operates within. As shown in FIG. 5, Merlin is shown within a three-tiered client/server architecture comprising user application layer 38, Merlin 40, and data sources (or virtual object databases) layer 56. Multiple heterogeneous data sources can reside either on the same or on distributed computer platforms. It also needs to be stressed that multiple user applications can utilize Merlin with customized object models in order to retrieve personalized views of the integrated data as illustrated by FIG. 5). *Hampstead et al* does not teach explicitly a script generation engine configured to produce a target specific set of instructions for a specified application component based upon a mapping of at least one of semantic models in repository. However, *Goiffon et al* teaches (figure 2, column 5, lines 19-25, processes can be used to develop Plans, wherein a Plan is a developed script that may be executed by a script engine to complete the task described by an associated process. Plans include a designation of the software modules that are to be used as input and output parameters of the task. A Plan may be provided to a script engine for execution either immediately after the plan is created from a process, or on a scheduled basis (column 40, lines 15-22, the above-described method can better be understood by example. Assume a Plan element has already been created from the process "WRAP_XOPEN_ACTX". This Plan element contains a list of user-supplied parameters in the "PlanCommandLineArgs" that was generated pursuant to the method described in FIG. 11. When this Plan is to be executed, the script is generated using the steps described in FIG. 12). Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention was made to incorporate semantic model with script generation. The modification would have been obvious because one of ordinary skill in the art would have been motivated to combine teaching into generating script for targeted machine according to semantic model in reference to achieve better performance and make it error free.

Regarding claims 2 and 6

Goiffon et al teaches,

semantic models comprises a listing of component relationships, target platform requirements and platform neutral installation instructions (column 13, lines 17-31, in the preferred embodiment, the Element Repository 220 is loaded with the Element Repository Model (ERM) 226. The ERM is an object model which defines objects within ER 220 used to store the element types, relationship types, elements, and relationships. Within the ERM, the model that defines the various element types is called the Element Inventory Schema (EIS) 103, as discussed above. This model is installed in the Element Repository at system installation time, but may be varied throughout the life of the system. The model definition may be specific to a particular Object Management System. The Element type definitions within EIS 103 provide the templates used during element creation, and define the type of information contained in, the attributes associated with, and the relationships existing between, the elements).

Regarding claims 3 and 7

Goiffon et al teaches,

component relationships comprises at least one component relationship selected from the group consisting of a containment relationship, a usage relationship, a contradiction relationship, and an equivalence relationship (column 39, lines 29-40, next, Plan Wizard creates an element of type "Plan" in Element Inventory 102 for the newly-created Plan. The "CommandLineArg" attribute for this element is set to the value contained in variable "PlanCommandLineArgs". This is illustrated by Step 1136. A relationship of type "Plan.Performs.Process" is created between the newly-created Plan element and the original, user-selected Process element, as shown in Step 1138. Additional relationships of type "Plan.Uses.Asset" are created in the Element Inventory between the Plan element and each selected Asset element, with the usage attribute of these relationships set to "uses_asset" as shown in Step 1140).

Regarding claim 4

Goiffon et al teaches,

a Web services interface to repository configured to permit remote access to repository (column 12-13, lines 66-67 and lines 1-15, AIM Server 214 includes Element Repository (ER) 220, which is the repository that stores and manages persistent objects (elements). ER 220 may be implemented across multiple hosts interconnected by a remote interface. In the preferred embodiment shown in FIG. 2, the ER is implemented using the Unisys Universal Repository (UREP) commercially available from the Unisys Corporation, although other types of repositories could be utilized including, but not limited to, a Microsoft Repository commercially available from the Microsoft Corporation. Unisys Universal Repository (UREP) is a fully object-oriented repository for providing access to, concurrent sharing of, and immediate update support of all objects stored within the repository. For more information on the UREP system from Unisys, see the UREP Technical Overview, Document Number 8807 6971-000 available from the Unisys Corporation, and which is incorporated herein by reference in its entirety).

Regarding claims 8 and 13

Hampstead et al teaches,

retrieving a semantic model for the application component from a communicatively coupled repository of semantic models (column 8, lines 14-25, the purpose of the object model information within the data source access support component of the knowledge base is to define the objects and attributes for which a specific data source will be providing the data. When defining the objects and the attributes, the data source mappings (i.e., the specific data source table and field from which the data is retrieved) must be provided, along with any join information if an object is created from data retrieved from multiple tables. The Merlin knowledge base toolkit 60 takes the information entered by the user into the graphical user interface (GUI) 74, translates it into the necessary ODL syntax (including mapping extensions), and stores it in the appropriate flat file, such as files 102 and 202.

Accordingly, data source access object model definition is provided by the knowledge base toolkit 60);

determining from semantic model, a set of dependent components required to be present in the specific target platform (column 10, lines 5-18, a third translation entails a conversion of MDBC Semantic Model to ODBC Catalog Information, identified by reference numeral 72 in FIG. 6. The ODBC standard specifies that catalog information (i.e., meta-data) should be returned in a specific manner via specific functions. The MDBC component returns its semantic model (i.e., meta-data) in a proprietary structure format defined in its Object Request Broker (ORB) Interface Definition Language (IDL) file. In order to convert the meta-data from the MDBC semantic model structure to the ODBC catalog structure, several algorithms were created. Multiple algorithms were needed because ODBC requires multiple method calls to return all the information that MDBC returns in its semantic model structure);

further determining from semantic model a set of resource requirements required to be met by the specific target platform (column 8, lines 1-13, more particularly, Merlin 40 relies on its knowledge bases 82, 84 and 86 in order to provide domain-specific details that are used at run-time. The two components that require knowledge base information are the integrator support component 76 and the data source access support component 78. The knowledge bases for the integrator support component 76 and the data source access support component 78 are comprised of a number of different types of knowledge. The Merlin knowledge base toolkit (KB Toolkit) 60 enables the capability to relatively easily enter and/or modify the various knowledge pieces in the appropriate knowledge bases 82, 84, and 86. The collective Merlin knowledge base is made up of a series of flat files. Hampstead et al does not teaches explicitly requirements into platform specific instructions in a platform specific installation script. However, *Goiffon et al* teaches, (column 14, lines 53-59, Import Element Types: This service reads element types from a file and writes them into the EIS 103 in a predetermined format, which in the preferred embodiment is AL format, and is shown by dashed line 227. This service is called by scripts that execute on the Script Server 218. The element types are installed at initialization time, and may be updated as desired during the life of a system).

Regarding claims 9 and 14

Goiffon et al teaches,

yet further determining from semantic model a set of platform neutral installation operations (column 7, lines 26-38, this meta-data describes, among other things, the location of, and the type of, data or code that is stored within the respective component or module residing elsewhere within the system. This meta-data also describes the various relationships that the respective data or code module has with other data and/or code modules. In this manner, the Element Inventory 102 serves as an index which points to, and describes, the various data and code resources used to perform the functions of the particular Information Technology (IT) platform which utilizes the Object Management System 100. This index provides a mechanism which allows a very large body of reusable code and data components to be readily managed using automated tools, and further provides users with an efficient means of understanding the complex relationships and interdependencies existing between the code and data components) ; and,

further mapping set of platform neutral installation operations into platform specific instructions (column 45, lines 47-52, includes relationship means for storing asset relationship indicators each for associating a respective one of said plans to a respective one of said asset objects for mapping said respective one of said plans to the software module described by said respective one of said asset objects).

Regarding claims 10, 11, 15 and 16

Hampstead et al teaches,

identifying a set of dependent components for the application component (column 10, lines 5-18, a third translation entails a conversion of MDBC Semantic Model to ODBC Catalog Information, identified by reference numeral 72 in FIG. 6. The ODBC standard specifies that catalog information (i.e., meta-data) should be returned in a specific manner via specific functions. The MDBC component returns its semantic model (i.e., meta-data) in a proprietary structure format defined in its Object Request Broker (ORB) Interface Definition Language (IDL) file. In order to convert the meta-data from the MDBC semantic model structure to the ODBC catalog structure, several algorithms were created. Multiple algorithms were needed because ODBC requires multiple method calls to return all the information that MDBC returns in its semantic model structure); and, further identifying a set of sub-dependent components for at least one of dependent components (column 10, lines 19-25, “also according to fig 6…database 156 and 256).

Regarding claims 12 and 17

Hampstead et al teaches,

the step of computing an composite set of resource requirements for the application component and for set of dependent components (figures 2-6).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anil Khatri whose telephone number is 571-272-3725. The examiner can normally be reached on M-F 8:30-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, Wei Zhen can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

***

ANIL KHATRI
PRIMARY EXAMINER